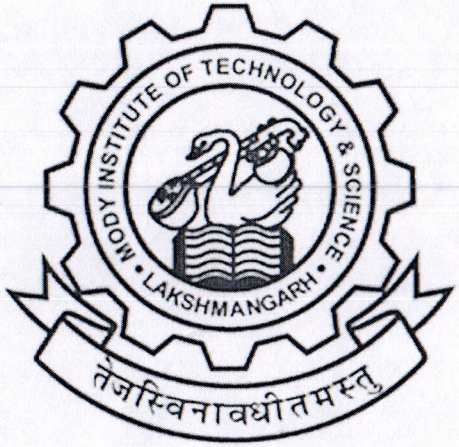
Title: "Digital Image Steganography using LSB, and Color Models"

Mody Institute of Technology and Science, Lakshmangarh
Faculty of Engineering and Technology



Supervisor:
Prof.(Dr.) S.K.Lenka
HOD, IT

Research Scholar:
Mamta Jain
En.No.070764

Title: "Digital Image Steganography using LSB, and Color Models"

Mody Institute of Technology and Science, Lakshmangarh
Faculty of Engineering and Technology

Supervisor:
Prof.(Dr.) S.K.Lenka
HOD, IT

Research Scholar:
Mamta Jain
En.No.070764

Contents:

- Introduction - Data Security
- Cryptography and Steganography Basics
- Categorization of Steganography Techniques
- Image Steganography Techniques & Block Diagram
- Applications of Image Steganography
- Literature Survey :
  - LSB Cipher Work
  - LSB Array Work
  - Color Models
- Proposed Research Work
- References
- Publications

Introduction

3

Introduction: Data Security

- Information and communication technology has grown rapidly.
- Internet is the most popular communication medium now a days.
- Message transmission over the internet is facing some problems such as data security.
- Need of secure secret communication methods for transmitting data over the internet.
- Security is the collection of tools designed to protect data and to prevent from hackers.
- Two techniques are used cryptography and steganography.

4

Cryptography v/s Steganography Basics

- Steganography and Cryptography are closely related
- The difference is in their goals...
  - Cryptography: Although encrypted and unreadable, the existence of data is not hidden-secret writing
  - Steganography: No knowledge of the existence of the data-covered writing
- Steganography and Cryptography can be used together to produce better protection of secret data

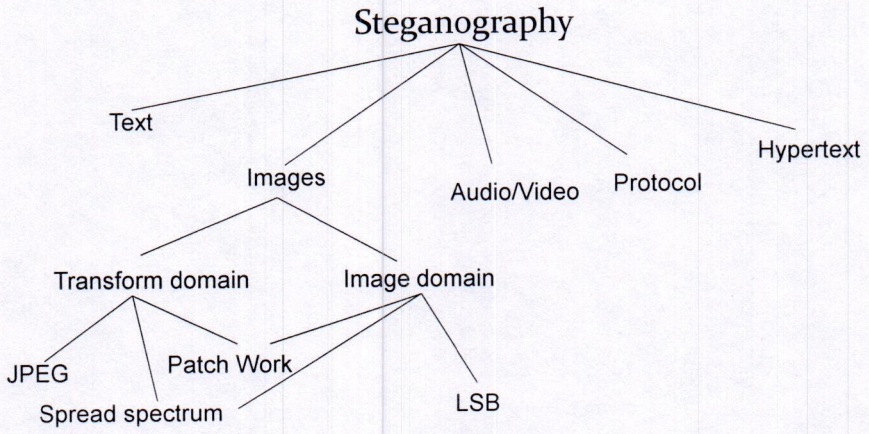
5

Cipher Text

- In cryptography, **cipher text** (or **cyphertext**) is the result of encryption performed on secret plaintext using an algorithm, called a cipher.
- Cipher text is also known as encrypted or encoded information.
- Decryption: The inverse of encryption, is the process of turning cipher text into readable form.

6

Categorization of Steganography Techniques



7

Overview of Steganography:

- **Text steganography -** A method is to hide a secret message in every nth letter of every word of a text message using digital files. It is not used very often since text files have a very small amount of redundant data.
- **Image Steganography -** Images are the most popular cover objects for steganography especially on the Internet, gives the large amount of redundant bits present in the digital representation of an image.
- **Audio steganography/Video Steganograhy-** Similar techniques as image files. One different technique unique to audio steganography is masking, which exploits the properties of the human ear to hide information unnoticeably.
- **Protocol Steganography-** The technique of embedding information within messages and network control protocols used in network transmission.
- **Hypertext Steganography-** Modified approach for text steganography based on HTML tags and attributes. As HTML is rich in tags and its attributes, easily communicated in the internet. By hiding the secret data inside the source code of HTML, text steganography can be easily achieved.[1]

8

Image Steganography Techniques :

1. Image Domain :

- Embed messages in the intensity of the pixels directly.
- Image domain techniques encompass bit-wise methods that apply bit insertion and noise manipulation.
- The image formats that are most suitable for image domain steganography are lossless and the techniques are typically dependent on the image format.

2. Transform Domain :

- Images are first transformed and then the message is embedded in the image.
- Independent of the image format and the embedded message may survive conversion between lossy and lossless compression.

Patch work:

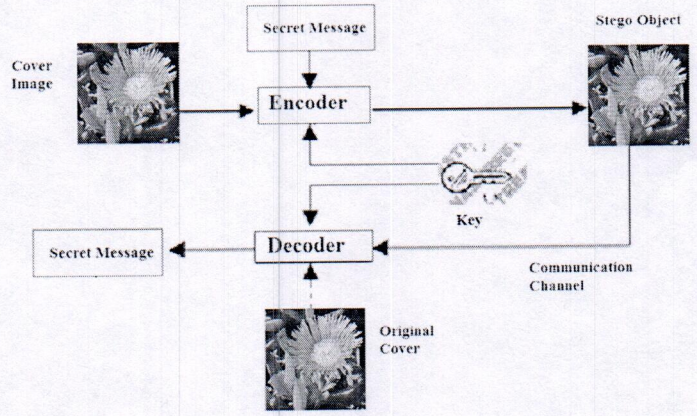
- Uses redundant pattern encoding to embed a message in an image.

Spread Spectrum:

- Hidden data is spread throughout the cover-image making it harder to detect.[1]

9

Image Steganography: Block Diagram



10

Applications of Image Steganography:

- Confidential communication and secret data storing
- Protection of data alteration
- Access control system for digital content distribution
- Media data base systems
- E-forensics steganography system for secret information retrieval
- Defense system

11

Literature Survey

12

Literature Survey:

LSB Cipher Work:

1. **A New Steganography Approach for Image Encryption Exchange by Using the LSB Insertion: [1]**

- This approach uses the Least Significant Bits (LSB) insertion to hide data within encrypted image data randomly.
- The method offers a good concealment for data in the encrypted image, and reduces the chance of the encrypted image being detected.
- The hidden data will be used to enable the receiver to reconstruct the same secret information after extracting it

Technique Used:

- The secret key
- The encrypted image represents the transformed image after encryption.
- The numbers of horizontal and vertical blocks represent the hidden information.
- The result is the mixed data; Encrypted image that includes the mixing data.

13

Data Mixing (Sender side) : ALGORITHM MIXING_DATA (Contd…)

1: Load ciphered image
2: Input key
3: Convert HNB into HNB_bitArray
4: Convert VNB into VNB_bitArray
5: seed = |Hash Function (Key)|
6: Randomize using seed
/* Replace bits of the HNB_bitArray with least significant bits of randomly selected pixels */
7: For i = 0 to length of HNB_bitArray – 1
7.1. r = rnd // r between 1 and # of image pixels -1
7.2. least bit of pixel r = HNB_bitArray(i)
/* Replacement of the i^{th} bit of HNB_bitArray with the least significant bit of the r^{th} pixel of the image */ /* Replace bits of the VNB_bitArray with least significant bits of randomly selected pixels */
8: For i = 0 to length of VNB_bitArray – 1
8.1. r = rnd // r between 1 and # of image pixels -1
8.2. least bit of pixel r = VNB_bitArray(i) /* Replacement of the i^{th} bit of VNB_bitArray with the least significant bit of the r^{th} pixel of the image */ END MIXING_DATA
Input: Ciphered image, Key Output: Mixed data

14

Data Extracting (Receiver side): **(Contd…)**

A user knows the key is the only one who can extract this information since the insertion positions depend on the secret key.

ALGORITHM SPLITTING_DATA:

1: Load mixed data
2: Input key
3: seed = |Hash Function (Key)|
4: Randomize using seed /* Extract bits of the HNB_bitArray */
5: For i = 0 to length of HNB_bitArray – 1
5.1. r = rnd // r between 1 and # of image pixels -1
5.2. HNB_bitArray(i) = least bit of pixel r /* Extract the i^{th} bit of HNB_bitArray from the least significant bit of the $(r)^{th}$ pixel of the image */
6: Convert HNB_bitArray into HNB /* Extract bits of the VNB_bitArray */
7: For i = 0 to length of VNB_bitArray – 1
7.1. r = rnd // r between 1 and # of image pixels -1
7.2. VNB_bitArray(i) = least bit of pixel r /* Extract the i^{th} bit of VNB_bitArray from the least significant bit of the $(r)^{th}$ pixel of the image */
8: Convert VNB_bitArray into VNB
END SPLITTING _DATA
Input: Mixed data, Key Output: Ciphered image, HNB, VNB

15

2. Steganography –using a Double Substitution Cipher:[3]

- In this paper author used algorithm called SWAIN-LENKA double substitution cipher algorithm to encrypt the secret information
- The encrypted information is embedded into the cover image in the following manner.
- In the first byte the 7th bit position, in the second byte the 8th bit position, in the third byte the 7th bit position, in the fourth byte the 8^{th} bit position and so on.
- This process is continued till the end of the encrypted data.
- Authors get two levels of security.

Algorithm Used:

Step1- Convert the cover image to binary
Step2–Convert the secret text to binary.
Step3 – Make sure that the length of cover image is at least 8 times than that of the secret text.
Step4–Apply *SWAIN-LENKA substitution cipher* algorithm to encrypt the secret information to get the cipher .
Step5 –Embed the cipher into the cover image.
Step6 – Now send the resultant image to receiver
Step7 – Receiver retrieves the bits from each byte from the different positions as he knows the positions and applies the *SWAIN-LENKA substitution cipher* algorithm to decryption.

16

THE SWAIN-LENKA DOUBLE SUBSTITUTION CIPHER (Contd...)

Encryption:

Table 1: First step of Encryption

a b c d e f g h i j k l m n o p q r s t u v w x y z
s w a i n o f g m r b c d e h j k l p q t u v x y z

Table 2: Second step of Encryption

a b c d e f g h i j k l m n o p q r s t u v w x y z
c d g h i j m q r s v w x y z l e n k a o f b p u t

The first step output is input to this step.

Decryption:

- The decryption process at the receiver is also two steps and the reverse of encryption.

17

3. A Technique for secure communication Using Message Dependent Steganography:[4]

- Author use substitution cipher called two square reverse to encrypt the secret information.
- The cipher text of the secret information is embedded into the carrier image in LSB minus one position of some selected bytes.
- The byte selection is done depending on the bit pattern of the secret information.
- The embedding locations are dependent on the secret message.
- So the intruder will face difficulties to locate the bits.

Algorithm Used:

Step1 - Convert the cover image to binary

Step2 - Apply Two Square Reverse to encrypt the secret information. Now got the cipher.

Step3 - Convert the cipher text to binary.

Step4 - Make sure that the length of cover image is sufficient enough to conceal the cipher text.

Step5 - Embed the cipher into the cover image.

Step6 - Now send the resultant image to receiver

Step7 - Receiver applies the reverse process what sender has done and gets the hidden information.

18

Selection of bytes as per the embedding process (Contd...)

Carrier File Byte	Operation
Byte A	Embed(1)
Byte B	skip
Byte C	Embed(1)
Byte D	skip
Byte E	Embed(0)
Byte F	Embed(0)
Byte G	Embed(1)
Byte H	skip
Byte I	Embed(0)
Byte J	Embed(1)
Byte K	skip
Byte L	Embed(1)
Byte M	skip
So on	

19

4. Steganography using the Twelve Square Substitution Cipher and an Index Variable:[5]

- Authors proposed a technique for secret communication
- Encrypt the secret message using cipher algorithm called twelve square substitution cipher.
- Embed the cipher text in the carrier image in 6^{th} and 7^{th} bit locations or 7^{th} and 8^{th} bit locations or 6^{th} and 8^{th} bit locations of the different pixels (bytes) of the carrier image depending on the value of an index variable.
- The index variable value can be 0 or 1 or 2.
- The index variable value will change from 0 to 1 or 1 to 2 or 2 to 0 after each embedding.
- The initial value of the index variable depends upon the length of the cipher text.
- The embedding locations are not same in all pixels, so it is a stronger approach.
- In authors earlier paper they represented a six square substitution cipher which includes only alphabets.
- This is extended to twelve square substitution cipher to include numerals and special characters.
- It is less susceptible to frequency analysis attacks.

20

Algorithm Used: (Contd...)

Step1 - Transform the carrier image to binary

Step2- Apply the Twelve Square Cipher to get the cipher text of the secret message.

Step3- Convert the cipher text to binary.

Step4- Make sure that the length of carrier image is sufficient enough to conceal the cipher text.

Step5- Embed the cipher into the cover image.

Step6- Now send the resultant image to receiver.

Step7- Receiver applies the reverse process what sender has done and gets the hidden information.

21

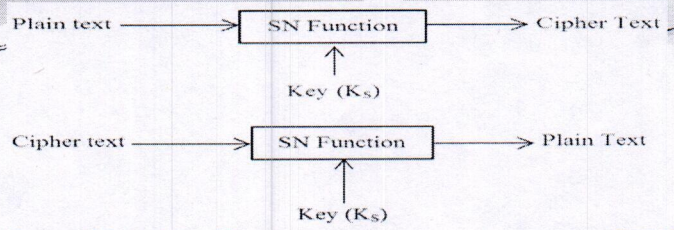
5. Image Steganography using Secret Key & Gray Codes[6]

- This algorithm takes image of any format like .jpeg, .gif, .bmp etc. as a carrier and converts it into .bmp format.
- The secret data bits are encrypted using gray codes and then this encrypted file is hidden in the LSB of carrier image.
- Use of key increases the security of the secret data.
- Gray Code is a form of binary that uses a different method of incrementing from one number to the next.

Table:

Decimal	Natural Binary	Gray Code
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100

22



Encrypting and Embedding Process: **(Contd…)**

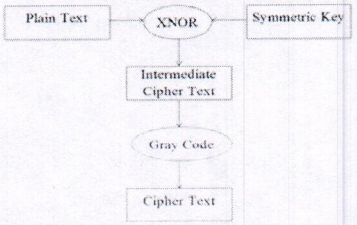
1. Take image of any format as an input.
2. Convert this input image into .bmp image format.
3. Take a data file (Text, Image, audio, Video) as an input.
4. Take a symmetric key as an input.
5. Extract every single byte of data file.
6. Apply SN function on it.
7. Take every single pixel of .bmp image.
8. Extract its RGB.
9. Hide information of data file like name, type, size etc. into the LSB of right most bottom pixels.
10. Hide the encrypted data bits into the LSB of RGB of the rest of the pixels.
11. Send this .bmp image file to the receiver.

23

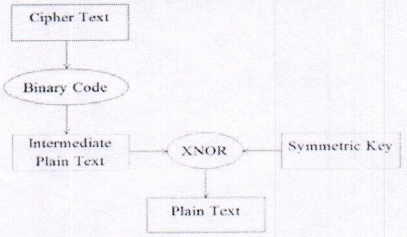
Extracting and Decrypting process: **(Contd…)**

1. Take the received .bmp image as an input.
2. Take symmetric key of 8 bit as an input.
3. Take every single pixel of input image.
4. Extract its RGB.
5. Extract data bits from LSB of RGB.
6. Arrange these extracted bits into group of 8 bits i.e. byte.
7. Apply SN function on these bytes.
8. Extract the data file information from the rightmost bottom pixels of the received image.
9. Create and view the encrypted hidden file.

SN Encryption Function



SN Decryption function



24

LSB Array Work:

- All the LSB bits of the different pixels as an array called LSB array.
- Mapped the encrypted message block to this LSB array, where maximum matched, there steganographed.
- The image in which the percentage of match is maximum can be used.

1. A Novel Steganography Technique By Mapping Words with LSB Array: [8]

- A new steganographic technique based on LSB array (least significant bit array) is proposed.
- Four LSB arrays such as LSB0, LSB1, LSB2 and LSB3 are defined.
- One of the four arrays is opted and formed depending upon the length of the secret message. For longer messages LSB3 array can be chosen.
- The different words of the secret message are mapped on the chosen array, where maximum match occurs, there steganographed and start indices are noted down.
- The length of each word and their start indices altogether are encrypted by RSA algorithm.
- The cipher text obtained is compressed and then embedded at some reserved location in the image.
- The extraction procedure is reverse of the embedding procedure.

25

Algorithm at Sender: (Contd...)

1. Input the image. Input the message.
2. Do not alter Pixels 1 to 3000, because image characteristics may be disturbed. Reserve the pixels from 3001^{th} to 34999^{th} for embedding word lengths and matched indices.
3. The pixels 35000^{th} to 95000^{th} (or up to the last pixel) are used to formulate the different arrays. The LSB0 array is formed by taking one least significant bit from each pixel i.e., the 8^{th} bits of the pixels. The LSB1 array is formed by taking two least significant bits of the pixels and so on . One of these arrays will be chosen for embedding the different words in it.
4. Do this for each word. Slide the binary word over the chosen binary array, where maximum matched steganograph there and note down the start index. In this way all the indices for the words are noted down.
5. Now compute the length of each word, Arrange length1, index1, length 2, index 2,.., length n, index n as another array, E.
6. Encrypt E by RSA algorithm, then compress it, say it is E_1. Find the length of E_1. Embed E_1 at two least significant bit locations of the pixels in the reserved area, i.e from 3001^{th} to 34999^{th} pixel. Now we got the stego image.
7. Stop.

Algorithm at Receiver:

1. Receive the stego image.
2. Formulate the selected array, i.e LSB0 or LSB1 or LSB2 or LSB3,
3. Retrieve the array E_1 from the two least significant bit locations of 3001^{th} to 34999^{th} pixels of the image. Decompress it, then decrypt it, thus we got E.
4. Calculate length1, index1, length2, indexe2,…, length n , index n from E.
5. Retrieve the words from the selected array with the help of these indices and lengths of the different words and form the message.
6. Stop.

26

2. LSB Array Based Image Steganography Technique by Exploring the Four Least Significant Bits:[9]

- A new LSB array based image steganographic technique using encryption by RSA algorithm is proposed.
- The four arrays, namely the LSB, LSB1, LSB2 and LSB3 are formulated separately by collecting the bits from the 8th(LSB),7th, 6th and 5th bit locations of the pixels respectively.
- The cipher text is divided into four blocks. The first block is mapped and slided over the LSB array and embedded at maximum matching portion of LSB array.
- Similarly the second, third and fourth blocks are embedded at maximum matching portion of LSB1, LSB2 and LSB3 arrays respectively.
- Where the blocks are embedded, the start indices are captured and embedded at a separate place in the image.
- The retrieving process at the receiver is the reverse of the sender.
- This LSB array concept is extended to have 4 arrays namely LSB, LSB1, LSB2 and LSB3, so that 4 different blocks of cipher text can be matched.
- The hiding capacity can be four times than that of the scheme in Juneja and Sandhu proposed a technique based upon LSB array.[7]

27

Algorithm at Sender: **(Contd...)**

Step-1: Input the secret message.

Step-2: Apply RSA algorithm to encrypt the secret message. Divide the cipher text into four blocks namely block0, block1, block2 and block3.

Step-3: Input the image. Transform it to binary.

Step-4: The first hundred bytes carries the image characteristics. Next to store the information like length of the blocks and the start index from the arrays where the blocks are matched, reserve from 101th byte upto 5000th byte. Then start from 5001th byte onwards upto the desired byte depending upon the length of your secret message.

Step-5: Map block0 with the LSB array. Start from the first location of LSB array, go on sliding the block0 on the LSB array, where it matches maximum, embed there and note down the start index in LSB array where steganographed. Say it is index1. Similarly find index2 fromLSB1 array after embedding block1 and so on

Step-6: Embed these four indices and the block lengths in the reserved pixels, using 7th and 8th bits of all those pixels.

Step-7: Stop

28

Algorithm at Receiver: **(Contd...)**

Step-1: Receive the stego image.
Step-2: Compute the LSB, LSB1, LSB2 and LSB3 arrays.
Step-3: Retrieve index0, index1, index2 and index3 from the reserved pixels.
Step-4: Retrieve block0 from LSB array from the location starting with index0.
Retrieve block1 from LSB1 array from the location starting with index1.
Retrieve block2 from LSB2 array from the location starting with index2.
Retrieve block3 from LSB3 array from the location starting with index3.
Step-5: Concatenate all these 4 blocks, thus we get the cipher.
Step-6: Apply RSA algorithm to decrypt it to get the plain text which is the secret message.
Step-7: Stop.

29

Color Models used in Steganography: RGB Channel

1. A Better RGB Channel Based Image Steganography Technique:[11]

- A new RGB channel based steganographic technique is used.
- The RSA algorithm is used for encryption and decryption.
- In a RGB image each pixel (24 bits) is having R-channel 8 bits, G-channel 8 bits and B channel 8 bits.
- Out of these three channels, the one which is having the maximum value summed over for all the pixels will be decided as the indicator channel.
- Out of the remaining two channels (say channel1 and channel2) in each pixel, one channel is used for embedding four bits of the cipher text, depending on its value satisfying one of the four defined conditions.
- At the receiver side the reverse operation will be performed to retrieve the message.
- In "RGB Intensity Based Variable-Bits Image Steganography " an algorithm is used where the indicator channel is randomly chosen, and 2 bits are embedded in a channel other than the indicator. [10]

30

Algorithm at Sender: (Contd...)

Step-1: Receive the secret message. Apply RSA Algorithm and convert it to cipher text.

Step-2: Receive the RGB Image. Convert to binary. The channel having maximum sum is the indicator channel. Four bit data is to be hidden in one of the channels other than the indicator, in the following manner.

Step-3: Suppose, the two channels other than the indicator channel are channel1 and channel2.

Step-4: Divide the cipher text into groups of four bits each.

Step-5: Take the next pixel, take the next four bits of cipher text. Do any of the following 4 sub steps.

(a). If (channel1 value $\leq$ 63 and channel2 value > 63) Embed the 4 bits of cipher in channel1 and set 8th bit of indicator channel to 0.

(b). If (channel1 value > 63 and channel2 value $\leq$ 63) Embed the 4 bits of cipher in channel2 and set 8th bit of indicator channel to 1.

(c). If (Channel1 value $\leq$ 63 and channel2 value $\leq$ 63) Then, If Channel1 value $\leq$ Channel2 value, Embed in channel1 and set 8th bit of indicator channel to 0. Otherwise Embed in channel2 and set 8th bit of indicator channel to 1.

(d). If Channel1 value > 63 and channel2 value > 63) do not embed in this pixel.

Step-6: If embedding of cipher text is not yet over go to step-5.

Step-7: Stop

31

Algorithm at Receiver: (Contd...)

Step-1: Receive the Stego Image. Transform to binary. Detect the indicator channel; this information is hidden in some reserved location of stego image. Detect the starting pixel and ending pixel locations with in which the cipher is embedded, these information is also hidden in the reserved location of the image.

Step-2: Suppose, the two channels other than the indicator channel are channel1 and channel2.

Step-3: Begin with start pixel (from where embedding continues)

Step-4: If both channel1 value and channel2 value are > 63 skip this pixel and go to step-5, otherwise do step 4 and go to step-5. If 8th bit of indicator channel is 0, take last four bits of channel1, otherwise take the last four bits of channel2. Concatenate these 4 bits to a variable named as CIPHER. Initially the variable CIPHER is blank.

Step-5: Take the next pixel. If it is the pixel following the end marker (it is the last pixel up to which embedding is done), go to step-6, otherwise go to step-4.

Step-6: Now apply RSA algorithm to decrypt the cipher text stored in the Variable CIPHER.

Step-7: Stop

32

2. A Novel Approach to RGB Channel Based Image Steganography Technique:[12]

- A novel approach to RGB channel based steganography technique is used.
- The RSA algorithm is used for encryption and decryption.
- In an RGB image, each pixel (24 bits) is having R-channel 8 bits, G-channel 8 bits and B-channel 8 bits.
- The image and cipher text is divided into 8 blocks each. One cipher block is allocated to be embedded in only one image block by a user defined sub key.
- Out of the three channels in each pixel of the image one is used as the indicator channel.
- The indicator channel for the different blocks is not the same.
- The other two channels (called data channels) are used for hiding cipher text bits in 4 least significant bit (LSB) locations.
- In a data channel 4 bits of cipher text can be embedded, if after embedding the change in pixel value is less than or equal to 7.
- The two LSBs of indicator will tell whether the cipher text is embedded in only one data channel or in both data channels.
- Retrieving can be done accordingly at the receiver.

33

The Algorithm at Sender: **(Contd...)**

- Divide the image into 8 blocks after leaving some reserved bytes (say, the initial 3099 bytes) at the beginning of the image. The 8 image blocks are B_0, B_1, B_2, B_3, B_4, B_5, B_6 and B_7.
- Encrypt the plain text by using RSA algorithm, thus the cipher text is obtained.
- Divide the cipher text into 8 blocks in the following manner.
- The cipher text comprises of so many bytes. Each byte is 8 bits. Keep together the first bits (most significant bits) of all the bytes, it becomes message block M0. Keep together the second bits of all the bytes and so on
- Enter the sub key K1, this sub key is for the allocation of message blocks to image blocks for possible embedding. This is a string of digits with a length of 8, formed by the digits 0 through 7, such that each digit occurs only once in the string. For example a sub key K1 = 05432167 means message block M0 is to be allocated to image block B0, message block M5 is to be allocated to image block B1 and so on.
- The second sub key K2, which tells about the indicator channels of all the 8 different image blocks, is to be calculated from the image blocks. From each block one indicator channel is calculated. One of the red, green and blue channels will be the indicator channel, whose sum over all the pixels in that block is maximum. Suppose the sequence is RBBRGBGR. R corresponds to 0, G corresponds to 1 and B corresponds to 2. Thus the sequence is 02201210, which is the sub key K2. Thus our key K= K1 K2 = 05432167 02201210, which is to be hidden in the reserved location of the image along with block length of image using two least significant bits in each byte.
- Suppose, I is the indicator channel and the other two channels are channel1 and channel2.

34

(Contd...)

- Now the different message blocks are allocated to different image blocks and in each image block the indicator is decided. Now the embedding in a block is done as follows.

a. Take the next pixel of the image block. Take 4 bits of the cipher text.

b. Compare 4 LSBs of channel1 with 4 bits of cipher text, if the difference is less than or equal to 7 then embed these 4 cipher text bits at those 4 LSBs of channel1. Take next 4 bits of cipher text, go to step(c). Otherwise (if the difference is greater than 7) with the same 4 bits of cipher text go to step(c).

c. Compare 4 least significant bits of channel2 with 4 bits of cipher text, if the difference is less than or equal to 7 then embed these 4 cipher text bits at those 4 LSBs of channel2. Otherwise do not embed in channel2, and those 4 cipher text bits are to be considered for the next pixel.

d. Go to step (a).

- For a pixel if embedding is done as per step (b) only, i.e data is embedded in channel1 only, then the two LSBs of its indicator will be set to 00. If for a pixel embedding is done as per step(c) only i.e., data is embedded in channel2 only, then the two LSBs of its indicator will be set to 01. If for a pixel embedding is done as per both the steps (b) and (c) i.e., 4 bits embedded in channel1 and 4 bits embedded in channel2, then the two LSBs of indicator will be set to 10. If in a pixel data is embedded neither in channel1 nor in channel2, then the last two LSBs of its indicator will be set to 11.
- Stop.

35

The Algorithm at Receiver: **(Contd...)**

- Retrieve the sub key K_1, sub key K_2, and the length of the blocks from the reserved location in the image.
- Suppose K_1= 05432167 and K_2= 02201210. In K_2, 0 means Red, 1 means Green and 2 means Blue. Thus K_2 = 02201210, means Red is the indicator channel in image block B_0, Blue is the indicator channel in image block B_1, Blue is the indicator channel in image block B_2, Red is the indicator channel in image block B_3, Green is the indicator in image block B_4, Blue is the indicator channel in image block B_5, Green is the indicator channel in image block B_6 and Red is the indicator channel in image block B_7.
- For each block identify the channel1 and channel2 as the indicator channel is identified. If Red is the indicator channel, then Green is channel1 and Blue is channel2. If Green is the indicator channel, then Red is channel1 and Blue is channel2. If Blue is the indicator channel, then Red is channel1 and Green is channel2.
- Declare the variables CIPHER.

  Declare the variable CIPHERi for i=0 to 7.

  For i=0 to 7 do the following.

  Take the image Block Bi. Start from the first pixel.

  a. Read the indicator of the pixel.

  b. If the last two bits of indicator are 00, retrieve the 4 LSBs of channel1 and append to the variable CIPHERi.

36

(Contd...)

c. If the last two bits of indicator are 01, retrieve the 4 LSBs of channel2 and append to the variable CIPHERi.

d. If the last two bits of indicator are 10, retrieve the 4 LSBs of channel1 and channel2 and append to the variable CIPHERi.

e. If the last two bits of indicator are 11, neither retrieve from channel1 nor retrieve from channel2 of that pixel.

f. Take the Next pixel, Go to 4(a).

- Concatenate all the cipher blocks, CIPHER1 to CIPHER7 to get the CIPHER.
- Now apply RSA decryption to CIPHER to get the plain text.
- Stop.

37

Proposed Research Work

38

Proposed Research Work:

1. **LSB Cipher used in Digital Image Steganography**
   - 1.1 A Hybrid Approach of Digital Image Steganography Using Four Box Substitution Cipher
   - 1.2 Secure Communication Using Sixteen Square Substitution Cipher and Message-Location Dependent Steganography.
   - 1.3 Image Steganography Using LSB Insertion with Different Coded Formats.

2. **Different Color Models used in Digital Image Steganography**
   Novel Image Steganography Technique Using Different Color Models

39

1.1 A Hybrid Approach of Digital Image Steganography using Four Box Substitution Cipher:

- In this paper we are proposing algorithm called reverse four box substitution cipher algorithm to encrypt the secret information
- The encrypted information is embedded into the cover image in LSB (least significant bit) position of some selected bytes.
- The byte selection is done depending on the secret message.
- So the intruder will face difficulties to locate the bits.
- The resultant image will be sent to the receiver and receiver should retrieve and then decrypt and get the secret information.
- We get two levels of security.

THE REVERSE FOUR BOX SUBSTITUTION CIPHER:

- Includes two steps of encryption.
- In the first step we divide the secret plain text including blank space into three-three characters.
- Reverse the three-three characters and get the cipher.
- The reverse four box substitution cipher technique is applied to get the second step cipher.
- This technique encrypts alphabets (upper case and lower case), digits and special characters.

40

First cipher text

Box1	Box3
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z @ ? a b c d e f g h i j k l m n o p q r s t u v w x y z	0 1 2 3 4 5 6 7 8 9 ` ~ ! # $ % ^ & * ( ) _ - + = { [}] ; : " ' \ | <, >. €/

Final cipher text **(Contd..,)**

Box2	Box4
J K L M N O P Q R S T U V W X Y Z @ A B C D E F G H I i j k l m n o p q r s t u v w x y z ? a b c d e f g h	7 8 9 ` ~ ! # $ % ^ & * (0 1 2 3 4 5 6 }] ; : " ' \ | <, >. €/) _ - + = { [

At the receiver, the decryption process is just the reverse of encryption. The second step of encryption will be the first step in decryption and the first step of encryption will be the second step of decryption.

THE EMBEDDING PROCESS: In few selected bytes the 8th bit position i.e LSB positions are to be embedded based on the bit pattern of the cipher text.

Carrier File Byte	Operation
Byte A	Embed(1)
Byte B	Skip
Byte C	Embed(1)
Byte D	Embed(0)
Byte E	Skip
Byte F	Embed(0)
Byte G	Embed(1)

41

(Contd...)

THE PROPOSED ALGORITHM:

Step1- Convert the cover image to bytes

Step2–Apply reverse four box substitution cipher algorithm to encrypt the secret information. Now we got the cipher.

Step3- Convert the cipher text to binary.

Step4- Make sure that the length of cover image is sufficient enough to conceal the cipher text.

Step5 –Embed the cipher into the cover image.

Step6 –Now send the resultant image to receiver

Step7 –Receiver applies the reverse process what sender has done and gets the hidden information.

42

THE EXPERIMENTAL RESULTS: (Contd…)

Figure1. The original image

Figure 3. The received image at the receiver

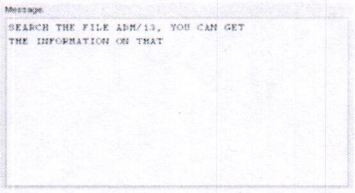


Figure 2.The secret information to be embedded

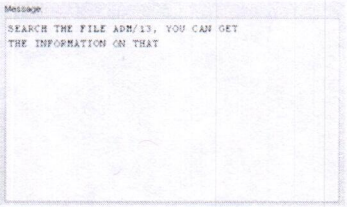


Figure 5. The retrieved information from the image at the receiver

43

Comparative Study: (Contd…)

- In previous papers people do steganography with LSB minus one position and LSB minus two positions.
- In these cases the stego image can be noticeably distorted.
- In this paper we are using LSB position, in some selected bytes.
- So that stego image can't be noticeably distorted.
- Embedding the cipher text not embedding the secret text directly.
- There are two levels of security in this proposed algorithm,
- first level of security it at the cryptography, Second will become the steganography.
- We are embedding based on the bits of the secret information; it becomes a stronger steganography.
- There is no chance for the intruder to identify the embedded pixels.
- If the intruder will try to hack by hit and trial method it is impossible for him to identify the pattern .
- It is dependent on the bits similarity and dissimilarity of the secret information.
- Everybody is using static way of calculating the embedded byte/pixel positions.
- We are using dynamic steganography.

44

References

45

References:

[1] T Morkel, JHP Eloff and MS Olivier, "An Overview of Image Steganography," in HS Venter, JHP Eloff, L Labuschagne and MM Eloff (eds), Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA2005), Sandton, South Africa, June/July 2005, http://mo.co.za/open/stegoverview.pdf

[2] Mohammad Ali Bani Younes, Aman Jantan, "A New Steganography Approach for Image Encryption Exchange by using the LSB insertion", IJCSNS International Journal of Computer Science & Network Security, Vol. 8, No. 6 , pp. 247-254, June 2008, Indexed: ProQuest, CiteFactor, EBSCO, Google Scholar.

[3] Gandharba Swain, S.K.Lenka, "Steganography-Using a Double Substitution Cipher", International Journal of Wireless Communications and Networking, Volume 2, Number 1, January-June 2010. pp.35-39. ISSN: 0975-7163, Indexed in: Ulrichs web, ebsco, doaj, open-jgate, Google scholar.

[4] Gandharba Swain, Saroj Kumar Lenka, " A Technique for Secure Communication using Message Dependent Steganography",Special issue of IJCCT, Vol. 2,No. 12,2010, Indexed in: Ulrichs web, ebsco, doaj, open-jgate, Google Scholar.

[5] Gandharba Swain, Saroj Kumar Lenka, "Steganography Using the Twelve Square Substitution Cipher and an Index Variable", Proceedings of Int Conference ICECT-2011, 7-9 April 2011. Kanyakumari, pp. 84-89 published in **IEEE Xplore** ISBN No: 978-1-4244-8679-37, doi: 10.1109/ICECTECH.2011.5941806, **Thomson Reuters, Scopus indexed.**

46

(Contd...)

[6] Sunny Dagar, Vinay Kumar, Yogendra Bagoriya, "Image Steganography using Secret Key & Gray Codes" , International Journal of Innovative Technology and Exploring Engineering(IJITEE) ISSN: 2278-3075,Volume-2, Issue-5, April 2013, Indexing:CiteFactor, Google Scholar, Scirus, getCITED, UK Index.

[7] Juneja, M., Sandhu, P.S. "Designing of Robust Steganography Technique Based on LSB Insertion and Encryption", IEEE International Conference on Advances in Recent Technologies in Communication and Computing, pp. 302–305 (2009), Indexed: Elsevier, Search, IET, **Thomson Reuters, Scopus indexed**.

[8] Gandharba Swain, Saroj Kumar Lenka, "A Novel Steganography Technique By Mapping Words with LSB Array", Signal and Imaging Systems Engineering, **Inderscience**, e-ISSN: 1748-0701, p-ISSN: 1748-0698, Indexed in: Google scholar, inspec, scirus, **Thomson Reuters, Scopus, SCI Indexed, 2012, Impact Factor =2.4.**

[9] Gandharba Swain, Saroj Kumar Lenka, "LSB Array Based Image Steganography Technique by Exploring the Four Least Significant Bits", Global Trends in Information Systems and Software Applications: 4th International conference Obcom2011, TN, India, **Communications in Computer and Information Science,** Vol. 270, Part II, pp.479-488, 2012 **Springer-Verlag** Berlin Heidelberg. doi: 10.1007/978-3-642-29216-3_52, ISBN 978-3-642-29215-6, e-ISBN 978-3-642-29216-, **Thomson Reuters, DBLP, Scopus.**

[10] Parvez, M.T., Gutub, A.A, "RGB Intensity Based Variable-Bits Image Steganography", Asia-Pacific Services Computing Conference, 2008. APSCC '08. IEEE Digital Object Identifier: 10.1109/APSCC.2008.105 , Publication Year: 2008 , pp.1322-1327, **Thomson Reuters, Scopus indexed.**

47

(Contd...)

[11] Gandharba Swain, Saroj Kumar Lenka, "A Novel Approach to RGB Channel Based Image Stegano-graphy Technique", Int. Arab Journal of e-Technology, Vol.2, Issue 4, pp.181-186, June 2012. print ISSN:1997-6364, Indexed in: Ulrichs web, ebsco, doaj, open-jgate, **DBLP indexed**.

[12] Gandharba Swain, Saroj Kumar Lenka, "A Better RGB Channel Based Image Steganography Technique", Global Trends in Information Systems and Software Applications: 4th International conference Obcom2011, TN, India,, **Communications in Computer and Information Science**, Vol. 270, Part II, pp.470-478, 2012 **Springer-Verlag** Berlin Heidelberg. DOI: 10.1007/978-3-642-29216-3_51, ISBN 978-3-642-29215-6, e-ISBN 978-3-642-29216-3.

[13] Gandharba Swain, Saroj Kumar Lenka, "A Technique for Secret Communication by Using a New Block Cipher with Dynamic Steganography", International Journal of Security and Its Applications, Vol. 6, No.2, pp.1-12, April 2012. ISSN: 1738-9976, indexed in: ebsco, proquest, Ulrich, doaj, openj-gate, cabell, ei compendex, (SERSC- South Korea), **Thomson Reuters, Scopus, Impact Factor =0.167.**

[14] Gandharba Swain, Saroj Kumar Lenka, Steganography Using Two Sided, Three Sided and Four Sided Side Match Methods", CSI Transactions on ICT (**Springer**), p-ISSN: 2277-9078, e-ISSN: 2277-9086, indexed in: Google scholar, OCLC, **Thomson Reuters, Scopus, DBLP, SCI Indexed.**

48

Publications:

1. Mamta Jain and S.K. Lenka **"A Hybrid Approach of Digital Image Steganography Using Four Box Substitution Cipher"**, AIMoC 2014, IEEE Kolkata Section, Jadavpur University, Communicated.
2. Mamta Jain, Saroj Kumar Lenka **"A Survey on Digital Image Steganography",** National Conference on Emerging Trend & its Application in Engineering (NCETAE 2011), Page no. 153-159, Dec 2011
3. Mamta Jain, Rohaila Naaz, Vibha Ojha and Anand Sharma**" Tongue Biometrics Authentication in Public System using Steganography",** International Conference on Technology and Management (ICTM 2011), Volume 1,Page no.22-25, June 2011, ISBN:978-93-81361-08-5.

49

THANKS

50